

Chapter 8

Mathematical and Business Functions

GOALS:

1. Students will be able to use Built-in Business Functions to solve financial problems.
2. Students will be able to create List and Combo boxes.

ESSENTIAL QUESTIONS

1. Why use Built-in Business Function instead of creating your own?
2. What are the main differences between List & Combo boxes?

LESSON

1. Math Operators

The following are the basic math operators built into the Visual Basic programming languages with corresponding code examples:

- Addition (+) -- For example, the line of code "4 + 5" returns 9.
- Subtraction (-) -- For example, the line of code "11 - 4" returns 7.
- Multiplication (*) -- For example, the line of code "3 * 7" returns 21.
- Float Division (/) -- For example, the line of code "10 / 3" returns 3.33333....
- Integer Division (\) -- For example, the line of code "9 \ 3" returns 9.
- Exponentiation (^) -- For example, the line of code "2^3" returns 8.
- Finding the remainder (Mod) -- For example, the line of code "7 Mod 3" returns 1.

2. Basic Math Functions

- Round(x,y) – Rounds x to y decimal places. If y is omitted, x is rounded to a whole number.
- Abs(x) – Returns absolute value of x.
- Sqr(x) – Returns square root of x.

3. Property Functions

- Sgn(x) – Returns the sign of x signified by 1 if x is positive, -1 if x is negative and 0 if the value of x is 0.
- Int(x) – Returns the whole number part of x.

4. Trigonometric Functions

- Sin(x) – Trigonometric sine of x (in radians).
- Cos(x) – Trigonometric cosine of x (in radians).
- Atn(x) – Trigonometric arctangent of x (in radians).

5. Exponent and Logarithmic Functions

- Exp(x) – Exponential function e to the power of x.
- Log(x) – Natural logarithm of x (base e).

Business Functions

To do this:	Use one of these functions:
Calculate depreciation.	DDB, SLN, SYD
Calculate future value.	FV
Calculate interest rate.	Rate
Calculate internal rate of return.	IRR, MIRR
Calculate number of periods.	NPer
Calculate payments.	IPmt, Pmt, PPmt
Calculate present value.	NPV, PV
<i>IsNumeric</i>	Returns a Boolean value indicating whether an expression can be evaluated as a number. Can be useful for validating input; but use with caution: if the expression contains the letter "E" or "D", the input argument could be interpreted as a number in scientific notation (thus returning True when you would expect the function to return False).

`Format(number, "format type")` – used to change the appearance of numeric data. *Number* – is a numeric variable or value and *format type*– is a predefined VB format.

Examples of different format types.

`lblAns.Caption = Format(8789, "General Number") → 8789`

`lblAns.Caption = Format(8789, "Currency") → $8,789.00`

`lblAns.Caption = Format(8789, "Fixed") → 8789.00`

`lblAns.Caption = Format(8789, "Standard") → 8,789.00`

`lblAns.Caption = Format(89, "Percent") → 8900.00%`

`lblAns.Caption = Format(8789, "Scientific") → 8.79E+3`

`lblAns.Caption = Format(1, "Yes/No") → Yes`

`lblAns.Caption = Format(0, "True/False") → False`

`lblAns.Caption = Format(1, "On/Off") → On`

`IsNumeric(argument)` – recognizes the \$ character and returns True if the characters after the \$ are numeric.

`Round(number, Decimal Places)` – Used to round numeric data to a specified number of decimal places.

`Pmt(rate, term, principal)` – *rate* is the interest rate per period, *term* is the total number of payments to be made, and *principal* is the amount borrowed. It would be used to determine monthly payments on a mortgage.

`PV(rate, term, payment)` – returns the present value of an investment. *Rate* – is the interest rate per period. *Term* – is the total number of payments to be made. *Payment* – is the amount invested per period. Example: `PV()` would be use to determine the financing a car.

`FV(rate, term, investment)` – returns what a series of equal payments invested at a fixed interest rate will be worth after a period of time.

List Boxes – allows the user to select a value from a set of values.

List Box Properties:

- List – stores a group of strings that make up the list
- Sorted – Displays the items in the list in Alphabetical order.
- ListCount - Property for the Combo and List boxes can be used only at run time to determine the number of items in the list.
- AddItem method – is used to add an item to the end of the combo box list at run time. Example:
lstControl.AddItem.Item
- Clear method – used to delete the contents of the combo box at run time. Example: *lstControl.Clear*

Combo Boxes – Collapsible list – allows user to select values from a set of values.

Combo Box Properties

- List – stores a group of strings that make up the list
- Sorted – Displays the items in the list in Alphabetical order.
- ListCount - Property for the Combo and List boxes can be used only at run time to determine the number of items in the list.
- AddItem method – is used to add an item to the end of the list at run time. Example:
cboControl.AddItem.Item
- Clear method – used to delete the contents of the list box at run time. Example: *cboControl.Clear*

ListCount Property for the Combo and List boxes can be used only at run time to determine the number of items in the list.

PROJECT 1 - Mathematical Functions

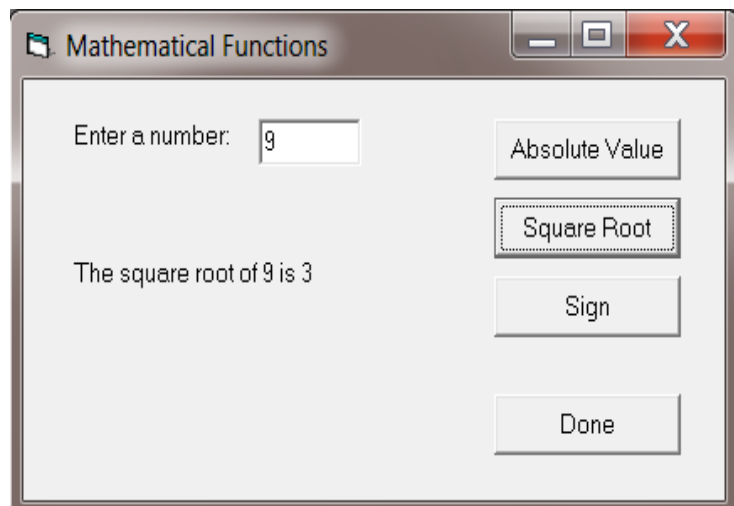
1. Create the following program
2. Answer the questions at the end of project

Option Explicit

```
Private Sub txtNumber_Change()  
    lblAnswer.Caption = ""  
End Sub
```

```
Private Sub cmdAbsoluteValue_Click()  
    Dim dblNumber As Double  
    Dim dblAnswer As Double
```

```
    If Not IsNumeric(txtNumber.Text) Then  
        MsgBox "Enter a numeric number."  
        txtNumber.Text = ""  
    Else  
        dblNumber = txtNumber.Text  
        dblAnswer = Abs(dblNumber)  
        lblAnswer.Caption = "The absolute value of " & dblNumber & " is " & dblAnswer  
    End If  
End Sub
```



```

Private Sub cmdSquareRoot_Click()
    Dim dblNumber As Double
    Dim dblAnswer As Double

    If Not IsNumeric(txtNumber.Text) Then
        MsgBox "Enter a numeric number."
        txtNumber.Text = ""
    Else
        dblNumber = txtNumber.Text
        If Sgn(dblNumber) = -1 Then
            lblAnswer.Caption = "There are no real square roots for " & dblNumber
        Else
            dblAnswer = Sqr(dblNumber)
            lblAnswer.Caption = "The square root of " & dblNumber & " is " & dblAnswer
        End If
    End If
End Sub

```

```

Private Sub cmdSign_Click()
    Dim dblNumber As Double
    Dim strAnswer As String

    If Not IsNumeric(txtNumber.Text) Then
        MsgBox "Enter a numeric number."
        txtNumber.Text = ""
    Else
        dblNumber = txtNumber.Text
        If Sgn(dblNumber) = -1 Then
            strAnswer = "negative"
        ElseIf Sgn(dblNumber) = 1 Then
            strAnswer = "positive"
        Else
            strAnswer = "a zero"
        End If
        lblAnswer.Caption = "The number " & dblNumber & " is " & strAnswer
    End If
End Sub

```

```

Private Sub cmdDone_Click()
    Unload Me
End Sub

```

Questions:

1. Why must you check the sign of the number in the square root function first before trying to take the square root of it?
2. How can you take a square root of a number without using the sqrt() function?
3. What is the purpose of the following statement in the above program?
If Not IsNumeric(txtNumber.Text) Then

PROJECT 2 - Round the Number

1. Create the following program
2. Answer the questions at the end of the project

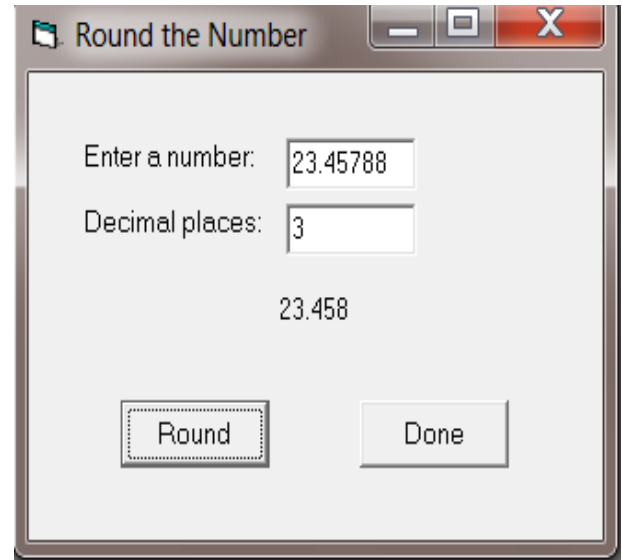
Option Explicit

```
Private Sub cmdRound_Click()  
    Dim dblNumber As Double  
    Dim intDecimalPlaces As Integer  
  
    dblNumber = txtNumber.Text  
    intDecimalPlaces = txtDecimalPlaces.Text  
    lblAnswer.Caption = Round(dblNumber, intDecimalPlaces)  
End Sub
```

```
Private Sub cmdDone_Click()  
    Unload Me  
End Sub
```

Questions:

1. When using the Round() function, what is the purpose of the two arguments (dblNumber & intDecimalPlaces)?



PROJECT 3 Installment Loan Payments

1. Create the following program
2. Answer the questions at the end of the project

Option Explicit

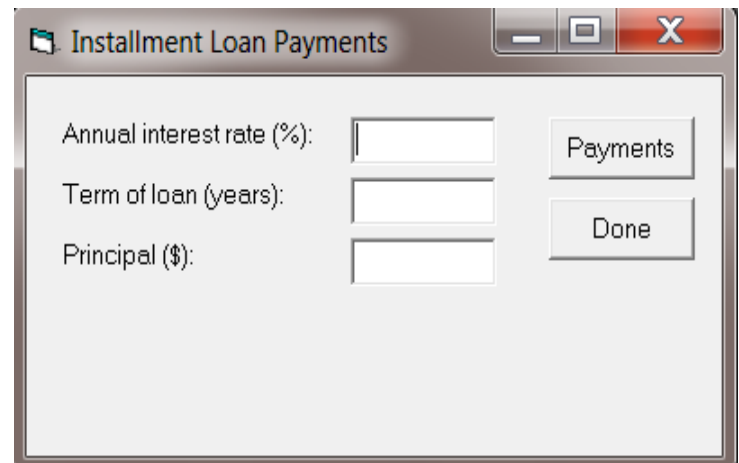
```
Private Sub txtInterestRate_Change()  
    lblAnswer.Caption = ""  
End Sub
```

```
Private Sub txtTerm_Change()  
    lblAnswer.Caption = ""  
End Sub
```

```
Private Sub txtPrincipal_Change()  
    lblAnswer.Caption = ""  
End Sub
```

```
Private Sub cmdPayments_Click()  
    Dim dblInterestRate As Double, intTerm As Integer, curPrincipal As Currency  
    Dim curAnswer As Currency
```

```
    dblInterestRate = txtInterestRate.Text  
    dblInterestRate = dblInterestRate/100  
    intTerm = txtTerm.Text  
    curPrincipal = txtPrincipal.Text
```



```

curAnswer = Pmt(dblInterestRate / 12, intTerm * 12, -curPrincipal)
lblAnswer.Caption = "The monthly payments for a loan of " & _
    Format(curPrincipal, "Currency") & " at " & Format(dblInterestRate, "Percent") _
    & " for " & intTerm & " years is " & Format(curAnswer, "Currency")
End Sub

```

```

Private Sub cmdDone_Click()
    Unload Me
End Sub

```

Questions:

1. Why was `dblInterestRate / 100`?
2. Explain the 3 arguments for the `Pmt()` function and their purposes.

PROJECT 4 How Much Can I Borrow

1. Create the following program
2. Answer the questions at the end of the project.

Option Explicit

```

Private Sub txtInterestRate_Change()
    lblAnswer.Caption = ""
End Sub

```

```

Private Sub txtTerm_Change()
    lblAnswer.Caption = ""
End Sub

```

```

Private Sub txtPayment_Change()
    lblAnswer.Caption = ""
End Sub

```

```

Private Sub cmdLoanAmount_Click()
    Dim dblInterestRate As Double, intTerm As Integer, curPayment As Currency
    Dim curAnswer As Currency

```

```

    dblInterestRate = txtInterestRate.Text
    dblInterestRate = dblInterestRate / 100
    intTerm = txtTerm.Text
    curPayment = txtPayment.Text
    curAnswer = PV(dblInterestRate / 12, intTerm * 12, -curPayment)
    lblAnswer.Caption = "You can borrow " & Format(curAnswer, "Currency") & _
        " if you plan on making monthly payments of " & Format(curPayment, "Currency") & _
        " at " & Format(dblInterestRate, "Percent") & " for " & intTerm & " years."
End Sub

```

```

Private Sub cmdDone_Click()
    Unload Me
End Sub

```

Questions:

1. What is the difference between PV() & PMT() function?

Project 5 - Watch Your Money Grow

1. Create the following programs
2. Answer the questions at the end of project

Option Explicit

```
Private Sub txtInterestRate_Change()
```

```
    lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub txtTerm_Change()
```

```
    lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub txtPayment_Change()
```

```
    lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub cmdFutureValue_Click()
```

```
    Dim dblInterestRate As Double, intTerm As Integer, curPayment As Currency
```

```
    Dim curAnswer As Currency
```

```
    dblInterestRate = txtInterestRate.Text
```

```
    dblInterestRate = dblInterestRate/100
```

```
    intTerm = txtTerm.Text
```

```
    curPayment = txtPayment.Text
```

```
    curAnswer = FV(dblInterestRate / 12, intTerm * 12, -curPayment)
```

```
    lblAnswer.Caption = "Your investment will be worth " & Format(curAnswer, "Currency") & _  
        " if you plan on making monthly investments of " & Format(curPayment, "Currency") & " at " & _  
        & Format(dblInterestRate, "Percent") & " interest for " & intTerm & " years."
```

```
End Sub
```

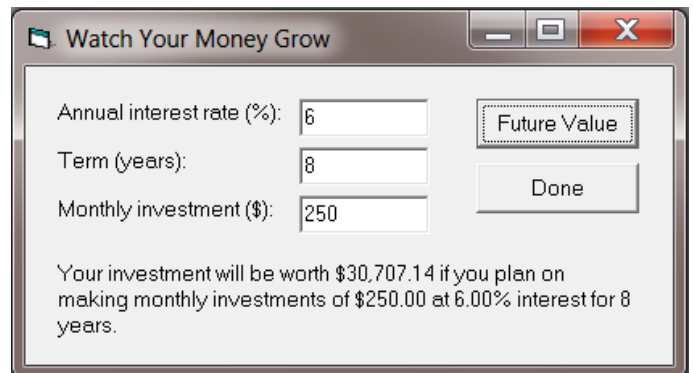
```
Private Sub cmdDone_Click()
```

```
    Unload Me
```

```
End Sub
```

Questions

1. What is the purpose of the FV() function?
2. Describe the three functions and their purposes?
 - a. PMT()
 - b. PV()
 - c. FV()



Project 6 Tuition Calculator

1. Create the following program.
2. Answer the questions at the end of project.

Option Explicit

```
Const strUndergrad As String = "Undergraduate"  
Const strGrad As String = "Graduate"  
Const strThesis As String = "Thesis and Dissertation"
```

```
Private Sub Form_Load()  
    lstCourseLevel.AddItem strUndergrad  
    lstCourseLevel.AddItem strGrad  
    lstCourseLevel.AddItem strThesis  
End Sub
```

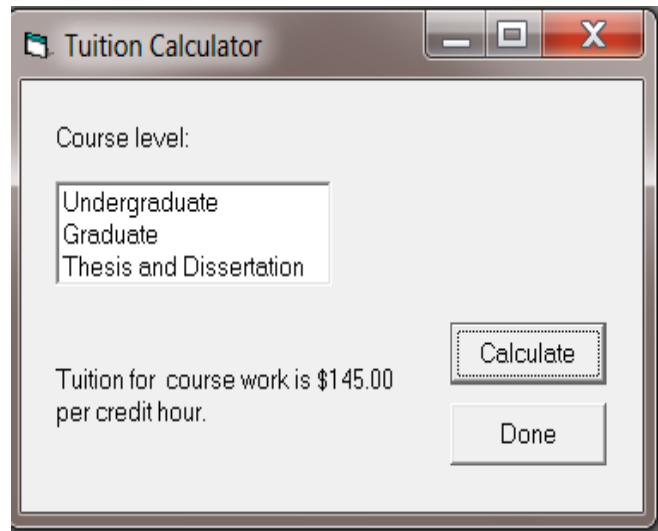
```
Private Sub lstCourseLevel_Click()  
    lblAnswer.Caption = ""  
End Sub
```

```
Private Sub cmdCalculate_Click()  
    Const curUndergraduatePerHour As Currency = 75  
    Const curGraduatePerHour As Currency = 145  
    Const curThesisPerHour As Currency = 145  
    Dim strCourseLevel As String  
    Dim curTuition As Currency  
  
    strCourseLevel = lstCourseLevel.Text  
  
    If strCourseLevel = strUndergrad Then  
        curTuition = curUndergraduatePerHour  
    ElseIf strCourseLevel = strGrad Then  
        curTuition = curGraduatePerHour  
    Else  
        curTuition = curThesisPerHour  
    End If  
  
    lblAnswer.Caption = "Tuition for " & strCourseLevel & " course work is " & _  
        Format(curTuition, "Currency") & " per credit hour."  
End Sub
```

```
Private Sub cmdDone_Click()  
    Unload Me  
End Sub
```

Questions

1. How do you add items to a list?
2. What code do you use to retrieve the selected item from the list?
3. Why did they use an extended if statement in the cmdCalculate button?



Project 7 Modify Tuition Calculator

1. Create the following program
2. Answer the questions at the end of the project

Option Explicit

Const strUndergrad As String = "Undergraduate"

Const strGrad As String = "Graduate"

Const strThesis As String = "Thesis and Dissertation"

Private Sub Form_Load()

 lstCourseLevel.AddItem strUndergrad

 lstCourseLevel.AddItem strGrad

 lstCourseLevel.AddItem strThesis

End Sub

Private Sub lstCourseLevel_Click()

 lblAnswer.Caption = ""

End Sub

Private Sub cboCreditHours_Change()

 lblAnswer.Caption = ""

End Sub

Private Sub cboCreditHours_Click()

 lblAnswer.Caption = ""

End Sub

Private Sub cmdCalculate_Click()

 Const curUndergraduatePerHour As Currency = 75

 Const curGraduatePerHour As Currency = 145

 Const curThesisPerHour As Currency = 145

 Dim intCreditHours As Integer

 Dim strCourseLevel As String

 Dim curTuition As Currency

 intCreditHours = cboCreditHours.Text

 strCourseLevel = lstCourseLevel.Text

 If strCourseLevel = strUndergrad Then

 curTuition = curUndergraduatePerHour * intCreditHours

 ElseIf strCourseLevel = strGrad Then

 curTuition = curGraduatePerHour * intCreditHours

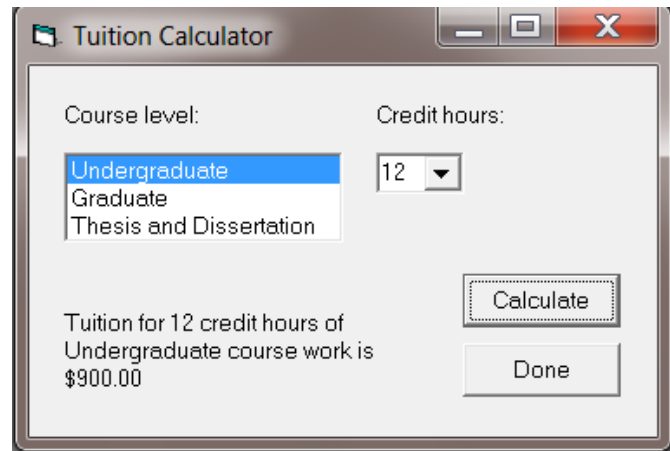
 Else

 curTuition = curThesisPerHour * intCreditHours

 End If

 lblAnswer.Caption = "Tuition for " & intCreditHours & " credit hours of " & _
 strCourseLevel & " course work is " & Format(curTuition, "Currency")

End Sub



```
Private Sub cmdDone_Click()
```

```
    Unload Me
```

```
End Sub
```

Questions

1. What part of the code was modified?
2. What effect did the modification have on the code?

Project 8 Modify Installment Loan Payment

1. Create the following program
2. Answer the questions at the event of the project.

Option Explicit

```
Private Sub cboInterestRate_Change()
```

```
    lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub cboInterestRate_Click()
```

```
    lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub cboTerm_Change()
```

```
    lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub cboTerm_Click()
```

```
    lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub txtPrincipal_Change()
```

```
    lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub cmdPayments_Click()
```

```
    Dim dblInterestRate As Double, intTerm As Integer, curPrincipal As Currency
```

```
    Dim curAnswer As Currency
```

```
    dblInterestRate = cboInterestRate.Text
```

```
    dblInterestRate = dblInterestRate / 100
```

```
    intTerm = cboTerm.Text
```

```
    curPrincipal = txtPrincipal.Text
```

```
    curAnswer = PMT(dblInterestRate / 12, intTerm * 12, -curPrincipal)
```

```
    lblAnswer.Caption = "The monthly payments for a loan of " & _  
        Format(curPrincipal, "Currency") & " at " & Format(dblInterestRate, "Percent") & _  
        " for " & intTerm & " years is " & Format(curAnswer, "Currency")
```

```
End Sub
```

Installment Loan Payments

Annual interest rate (%): 8

Term of loan (years): 30

Principal (\$): 100,000

Payments

Done

The monthly payments for a loan of \$100,000.00 at 8.00% for 30 years is \$733.76

```
Private Sub cmdDone_Click()
```

```
    Unload Me
```

```
End Sub
```

Questions

1. How was this program modified from the first program?
2. What is the difference between a TextBox and ComboBox

Project 9 Mortgage Analyzer

1. Create the following program
2. Answer the questions at the end of project.

Option Explicit

```
Private Sub Form_Load()
```

```
    Call ClearForm
```

```
End Sub
```

```
Private Sub lstWhatToCalculate_Click()
```

```
    lblEnterInformation.Visible = True
```

```
    lblRate.Visible = True
```

```
    cboInterestRate.Visible = True
```

```
    cboInterestRate.Text = 7
```

```
    lblTerm.Visible = True
```

```
    cboTerm.Visible = True
```

```
    cboTerm.Text = 15
```

```
If lstWhatToCalculate.Text = "The monthly payments of a mortgage" Then
```

```
    lblPayments.Caption = "&Loan amount ($):"
```

```
Else
```

```
    lblPayments.Caption = "&Maximum monthly payments ($):"
```

```
End If
```

```
lblPayments.Visible = True
```

```
txtPrincipal.Visible = True
```

```
lblAnswer.Caption = ""
```

```
End Sub
```

```
Private Sub cboInterestRate_Click()
```

```
    Call CheckValues
```

```
End Sub
```

'user clicked on new value in the combo box

```
Private Sub cboTerm_Click()
```

```
    Call CheckValues
```

```
End Sub
```

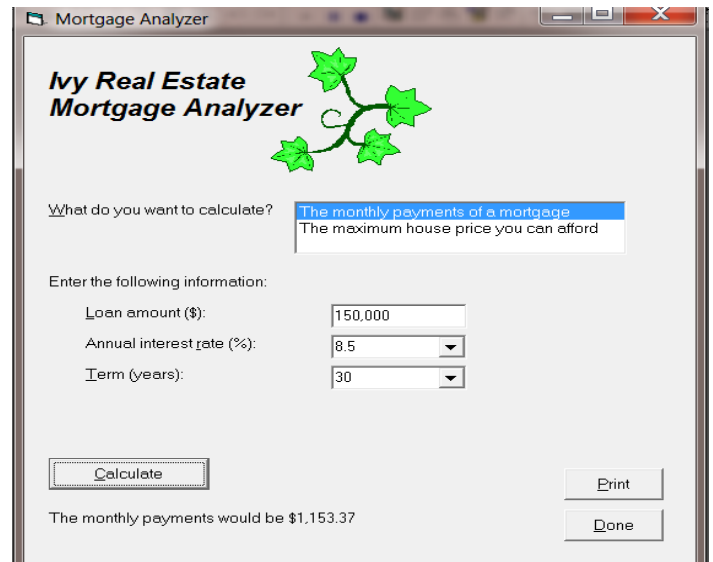
'user clicked on new value in the combo box

```
Private Sub cboInterestRate_Change()
```

```
    Call CheckValues
```

```
End Sub
```

'user typed a value in the combo box



```
Private Sub cboTerm_Change()           'user typed a value in the combo box
    Call CheckValues
End Sub
```

```
Private Sub txtPrincipal_Change()     'user types a value in the text box
    Call CheckValues
End Sub
```

```
*****
' Displays the Calculate command button if combo boxes
' and text boxes contain values greater than 0
'
' post: Calculate command button displayed if rate, term
' and payment values entered by user are greater than 0
*****
```

```
Sub CheckValues()
    If IsNumeric(cboInterestRate.Text) And IsNumeric(cboTerm.Text) _
        And IsNumeric(txtPrincipal.Text) Then
        cmdCalculate.Visible = True
    Else
        cmdCalculate.Visible = False
    End If
    lblAnswer.Caption = ""
End Sub
```

```
Private Sub cmdCalculate_Click()
    Dim dblInterestRate As Double, intTerm As Integer, curPrincipal As Currency
    Dim curAnswer As Currency

    dblInterestRate = cboInterestRate.Text / 100
    intTerm = cboTerm.Text
    curPrincipal = txtPrincipal.Text

    If lstWhatToCalculate.Text = "The monthly payments of a mortgage" Then
        curAnswer = Pmt(dblInterestRate / 12, intTerm * 12, -curPrincipal)
        lblAnswer.Caption = "The monthly payments would be " _
            & Format(curAnswer, "Currency")
    Else
        curAnswer = PV(dblInterestRate / 12, intTerm * 12, -curPrincipal)
        lblAnswer.Caption = "You can afford a house that is less than " _
            & Format(curAnswer, "Currency")
    End If
    lblAnswer.Visible = True
End Sub
```

```
Private Sub cmdPrint_Click()
    Me.PrintForm

    lstWhatToCalculate.Text = ""
    txtPrincipal.Text = ""
```

```
    Call ClearForm  
End Sub
```

```
Sub ClearForm()  
    lblEnterInformation.Visible = False  
    lblRate.Visible = False  
    cboInterestRate.Visible = False  
    lblTerm.Visible = False  
    cboTerm.Visible = False  
    lblPayments.Visible = False  
    txtPrincipal.Visible = False  
    cmdCalculate.Visible = False  
    lblAnswer.Caption = ""  
End Sub
```

```
Private Sub cmdDone_Click()  
    Unload Me  
End Sub
```

Run the program and experiment with it.

